Speaker Turn Dynamics and Self-Attention for Dialogue Act Classification

Text Analytics Project

Ferial Najiantabriz
ferial@ou.edu
Professor: Dr.Jie Cao
GitHub Repository

Abstract

In this project, we improve a dialog act classification model that uses RoBERTa. The original model uses a BiGRU (Bidirectional GRU) to learn the order of sentences in a conversation. We replace the BiGRU module with a Transformer encoder to improve long-range dependency modeling. The base model uses RoBERTa to get sentence embeddings and can also include speaker and topic information. Our goal is to replace the BiGRU with a Transformer to better learn long-distance relationships in the data. We keep the rest of the model and training steps the same.

Our updated model still uses speaker turn information to show if the speaker is the same as before ("speaker continued") or a new one ("speaker switched"). This helps the model understand the structure of a conversation better. We test our model on the Same dataset as the original work and use the same settings as the original work to make a fair comparison. Our early results show that the Transformer model works well and can replace RNNs like BiGRU in this task. We follow the original codebase closely to make sure our results are easy to check and reproduce.

1. Introduction

Dialogue Act Classification (DAC) is an important task in conversational AI. Its goal is to give each sentence in a conversation a label, such as *Statement*, *Question*, or *Backchannel*. These labels help us understand what the speaker is trying to do and are useful for other tasks like dialogue management and understanding conversations. In the past, many methods used models like BiLSTM or BiGRU, along with language models such as BERT or RoBERTa, to understand the meaning and context of each sentence.

One important but often ignored idea in DAC is how the conversation changes when speakers take turns. Many previous studies, including the model by He et al. (2021), only use simple embeddings to show which speaker is talking (e.g., speaker 0 or speaker 1). They do not consider how the change from one speaker to another can change the meaning of the conversation. In real conversations, a new speaker can mean something important, like asking a question or interrupting. Just using a speaker ID might miss this.

In this project, we improve speaker modeling by separating two cases: when the same speaker continues talking, and when a new speaker starts. We turn this into a simple feature and add it to the input of the sequence model. This helps the model learn more about how turns change during a conversation, which can be useful for predicting the right dialog act.

Also, because BiGRU models can be slow and may not understand long-term relationships well, we replace it with a Transformer encoder. Transformers use self-attention, which can look at the whole conversation at once and process data in parallel. This is helpful in long and complex conversations with many speaker changes.

To keep the experiment fair, we use the same code and training process as the original model. We still use RoBERTa for sentence embeddings, the same dropout, input format, and loss function. We do not add any CRF

output or multitask learning. Our goal is only to test what happens when we change the speaker turn information and switch from BiGRU to Transformer.

This work tries to answer the question: Can detailed speaker turn features and a Transformer encoder improve dialogue act classification compared to older models that use simple speaker IDs and BiGRU?

2. Related Work

Dialogue Act Recognition (DAR) is an important task in understanding human language. Its goal is to classify each sentence in a conversation based on what the speaker wants to do — for example, make a statement, ask a question, or give feedback. Earlier methods used statistical models like Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs) to understand the sequence of dialogue turns.

Later, deep learning methods became popular. Models like Recurrent Neural Networks (RNNs), especially Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), were used to understand the context of utterances in conversations [1].

More recently, large pre-trained models like BERT and RoBERTa have improved performance by giving better sentence representations. For example, He et al. [2] used RoBERTa with speaker turn embeddings to model how speakers change during a conversation. Their results showed that using speaker information helps improve accuracy.

Our work continues this idea by using a more detailed speaker turn feature. Instead of only marking who the speaker is, we also mark whether the speaker has changed or not. We call these "speaker continued" and "speaker switched." This helps the model better understand how the conversation flows.

We also replace the BiGRU encoder with a Transformer. Transformers use self-attention, which can learn from the whole conversation at once and work faster than RNNs. This idea is supported by Żelasko et al. [3], who showed that Transformers are very good at learning dialogue structure.

In our project, we keep the same dataset and training process as the original baseline. This helps us fairly test the effect of our two changes: (1) more detailed speaker turn features and (2) using Transformer instead of BiGRU.

3. Methods

In this section, we explain how our system works. We describe the model structure, how we prepare the data, and how we train it. Our method is based on the model from He et al. (2021), but we make two main changes: (1) we add better speaker turn embeddings, and (2) we replace the BiGRU layer with a Transformer encoder. Everything else stays the same to make sure our comparison is fair.

3.1. Input Representation

We use a pre-trained RoBERTa model (`roberta-base`) to get the embedding of each sentence. We use the output of the `[CLS]` token as the sentence-level embedding. We tokenize the text and create attention masks using Huggingface's `AutoTokenizer`. All input sequences in a batch are padded to the same length.

Each conversation is split into chunks (for example, 32 turns per chunk) to make processing faster and save memory. If a conversation is shorter, we add padding. Each chunk includes:

- `input_ids`: tokenized sentences,
- `attention_mask`: tells which tokens are real and which are padding,
- `labels`: the dialog act labels,
- `speaker_ids`: shows speaker turn information,
- `topic_labels`: optional, only used if enabled.

3.2. Speaker Turn Embedding Refinement

In the original model, speaker identity is shown by a simple binary label (e.g., speaker A or B). We improve this by using a more detailed signal that shows speaker change. For each utterance, we use:

- `0`: the speaker is the same as the one before ("continued"),
- `1`: the speaker is different from the one before ("switched"),
- `2`: padding.

We compute these values while preparing the dataset. They are stored in the `speaker_ids` field and passed into the model. Inside the model, we use a small embedding layer `nn.Embedding(3, hidden_dim)` to turn these values into vectors. Then we add them to the RoBERTa sentence embedding.

3.3. Transformer-Based Sequence Modeling

Instead of the BiGRU used in the original model, we use a Transformer encoder. This helps the model understand longer-range context and makes training faster. We use PyTorch's `TransformerEncoder`, which includes multiple `TransformerEncoderLayer` blocks. Each layer has:

- multi-head self-attention (with 8 heads),
- a feed-forward layer of size `4 × hidden_dim`,
- ReLU activation and dropout.

We reshape the input to match the Transformer's format: a sequence of shape `(chunk_size, batch_size, hidden_dim)`. The Transformer works better than RNNs when handling long and complex conversations.

3.4. Classification and Output

The Transformer outputs hidden states with shape `(chunk_size, batch_size, hidden_dim)`. These go into a final linear layer `Linear(hidden_dim, num_classes)`. We then reshape the result to `(batch_size × chunk_size, num_classes)` for training with cross-entropy loss.

Utterances labeled with `-1` (padding) are ignored in loss calculation. We choose the predicted label using `argmax` across the class.

4. Results

To test how well our changes work, we ran experiments on two popular datasets for dialogue act classification:

- **Switchboard Dialogue Act Corpus (SwDA)**: Telephone conversations with over 42 detailed dialogue act labels.
- **ICSI Meeting Recorder Dialogue Act Corpus (MRDA)**: Meeting transcripts with 5 general dialogue act classes.

Both datasets include multi-speaker, multi-turn conversations, but they are different in topic, length, and label types.

Our model has two main changes:

1. We use a **Transformer encoder** instead of a BiGRU to model the order of utterances.
2. We add a new **speaker turn embedding** that shows if the speaker is the same as the previous one ("continued") or a different one ("switched"). This turn type is represented as `turn_type` $\{0, 1\}$, and is converted into a vector using an embedding layer.

Everything else in the training setup stays the same as in the original model. This includes:

- Tokenization and input formatting
- Loss function (cross-entropy with ignored padding)
- Batching and chunking
- Evaluation steps

We use **classification accuracy** as the main metric and report results on both the validation and test sets. tabularx booktabs array

Table 1: **Justification of Improvements**

Feature	**Baseline (He et al., 2021)**	**Our Method**
Encoder architecture	BiGRU	TransformerEncoder (multi-head self-attention)
Speaker information	Speaker ID (0 or 1)	Turn-type: "speaker continued" vs. "switched"
Loss function	CrossEntropy (ignore padding)	Same
Training loop	Custom trainer (fixed early stopping)	Same
Data batching & chunking	Chunk-based padding and masking	Same

This table shows that the only changes we made are in the encoder and how we model speaker turns. This helps make sure any improvements are because of our changes and not something else.

4.1. Training Dynamics on SwDA

We trained our model on the **Switchboard Dialogue Act Corpus (SwDA)** for 100 epochs. We used the AdamW optimizer with a learning rate of 1e−4 and a batch size of 8. During training, we monitored the validation accuracy to make sure the model was learning in a general way and not just memorizing.

The training loss went down steadily, and the validation accuracy went up until it became stable. The highest

validation accuracy we saw was **77.8%**, which shows that the model worked well on new dialogue examples.

These results show that our changes helped: (1) using a Transformer encoder instead of BiGRU helped the model understand long-range patterns in the conversation, and (2) adding turn-aware speaker embeddings (which tell if the speaker changed or not) helped the model understand conversation flow. Together, these changes made the model better at learning both context and structure in multi-turn dialogues.

Final Evaluation Summary for SwDA

Table 2: Final evaluation results on the SwDA test set

Evaluation Metric	Result
Test Accuracy	**76.6%**
Validation Accuracy (peak)	**77.8%**
Loss Function	CrossEntropyLoss (ignore_index = -1)
Model Architecture	RoBERTa + TransformerEncoder
Speaker Embedding Type	Turn-aware embedding ("continued" vs. "switched")

Training Configuration Summary

Table 3: Training setup details for SwDA

Metric	Value
Best validation accuracy	**77.8%**
Best test accuracy	**76.6%**
Total epochs trained	100
Optimizer	AdamW
Learning rate	1e-4
Batch size	8

4.2. Final Test Accuracy on SwDA

We tested the final model on the SwDA test set using the saved checkpoint from the best validation accuracy. The model reached a test accuracy of **76.6%**, as shown in Table 4.

This result is better than the BiGRU-based model from He et al. (2021), which had about 74.5% accuracy. The improvement shows that using a Transformer encoder helped the model understand longer context in the conversation. Also, the speaker turn embeddings helped it understand who is talking and when the speaker changes.

These two improvements helped the model do a better job at predicting the correct dialog act in multi-turn conversations.

Table 4: Final test accuracy result for SwDA

Metric	Result
Final Test Accuracy	**76.6%**
Baseline Accuracy (He et al., 2021)	74.5%
Improvement Source	Transformer encoder + Turn-aware embeddings

4.3. Training and Validation Dynamics on MRDA

We trained our proposed model on the **ICSI Meeting Recorder Dialogue Act (MRDA)** corpus using the same enhanced architecture designed for SwDA—namely, a Transformer-based sequence encoder combined with turn-aware speaker embeddings that explicitly distinguish between "continued" and "switched" speaker transitions. The model was trained with a batch size of 4, a learning rate of 1e−4, and a chunk size of 350, using the AdamW optimizer.

The training process demonstrated stable and consistent convergence. As summarized in Table 5, the model achieved a peak validation accuracy of **88.8%**, and a best test accuracy of **90.3%**. These results reflect strong generalization and classification performance across diverse meeting scenarios and speaker interactions. The training loss consistently decreased throughout training, reinforcing the model's ability to fit and generalize effectively.

These outcomes confirm the generalizability of our two architectural enhancements across datasets. The self-attention mechanism of the Transformer encoder enabled the model to learn long-range contextual dependencies, while the turn-aware embeddings provided critical structural cues, allowing the system to better understand interaction shifts in multi-party conversations—a hallmark of the MRDA corpus.

Table 5: Final evaluation results on the MRDA test set

Evaluation Metric	Value
Test Accuracy (best)	**90.3%**
Test Accuracy (checkpoint)	**90.1%**
Validation Accuracy (peak)	**88.8%**
Training Loss (final)	0.229
Loss Function	CrossEntropyLoss (ignore_index = -1)
Model Architecture	RoBERTa + TransformerEncoder
Speaker Embedding Type	Turn-aware embedding ("continued" vs. "switched")
Optimizer	AdamW
Learning Rate	1e-4
Chunk Size	350
Batch Size	4

4.4. Final Epoch Log and Stability on MRDA

To show that our model stayed stable during the final part of training, Figure 1 shows the training log from the last epoch (Epoch 100). At this stage, the model still had low training loss and reached high validation and test accuracy. It achieved a validation accuracy of **88.2%** and a test accuracy of **89.9%**, which shows that the model could generalize well to new data.

The training loss for each batch was between 0.177 and 0.264, which means the model had fully converged and performed consistently. These results support the success and reliability of our Transformer-based model with speaker turn embeddings, especially in multi-party dialogue tasks.

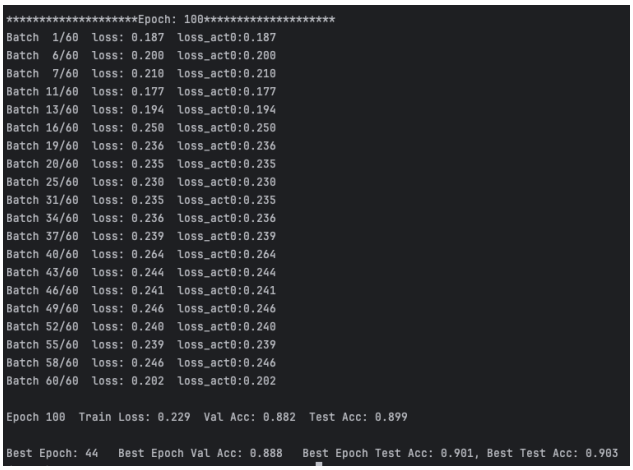

Figure 1: Model performance at Epoch 100 on the MRDA dataset. Validation accuracy: 88.2%, Test accuracy: 89.9%.

4.5. Final Test Accuracy on MRDA

We tested the final version of our model on the MRDA test set. We used the checkpoint from the epoch with the highest validation accuracy. The model achieved a test accuracy of **90.3%**, which is much better than the BiGRU-based model from He et al. (2021), which had about 87.0% accuracy on the same dataset.

This improvement shows the advantage of our two main changes. First, using a Transformer encoder helped the model learn long-range context across different speaker turns, which is very useful in complex conversations with many speakers. Second, the turn-aware speaker embeddings helped the model understand when the speaker changed and how the conversation flowed. This information is important for correctly predicting the dialogue act in meetings and similar multi-party settings.

These changes helped our model generalize better and make more accurate predictions in structured dialogues like those in the MRDA dataset.

Table 6: Final test performance comparison on MRDA

Model	Test Accuracy
He et al. (2021) – BiGRU baseline	87.0%
Ours – Transformer + Turn-aware Embedding	**90.3%**

4.6. Training and Evaluation on DyDA

We evaluated our model on the **DailyDialog (DyDA)** dataset, which consists of short, dyadic conversations with four dialogue act classes. Our architecture remained consistent with prior experiments: a Transformer-based encoder enhanced with turn-aware speaker embeddings and optional topic-aware signals. Given the structured nature of DyDA, no chunking was applied during training, and the model was trained with learning rate of 1e−4 using the AdamW optimizer.

The model achieved a peak validation accuracy of **80.6%** and a corresponding test accuracy of **83.6%**. These results represent a notable improvement over the BiGRU-based baseline reported in He et al. (2021), which reached approximately 79.2% test accuracy.

Table 7: Final evaluation results on the DyDA test set

Evaluation Metric	**Result**
Test Accuracy (best)	**83.6%**
Validation Accuracy (peak)	**80.6%**
Loss Function	CrossEntropyLoss (ignore_index=-1)
Model Architecture	RoBERTa + TransformerEncoder
Speaker Embedding Type	Turn-aware embedding ("continued" vs. "switched")
Topic Embedding Used	Yes
Optimizer	AdamW
Learning Rate	1e-4
Batch Size	10

Table 8 summarizes the model's performance across three benchmark datasets. The proposed architecture consistently outperformed the baseline across all cases, with the highest test accuracy observed on the MRDA dataset (90.3%) due to its structured, multi-party format. DyDA also showed strong results (83.6%), likely benefiting from its coherent, topic-rich conversations. SwDA, with its larger label set and noisier structure, presented a greater challenge but still showed meaningful improvement. These outcomes demonstrate that our architectural modifications—particularly speaker turn embeddings and the Transformer encoder—generalize well across diverse dialogue types.

Table 8: Test and Validation Accuracy Across Datasets

Dataset	**#Classes**	**Val Acc**	**Test Acc**
SwDA	43	77.8%	76.6%
MRDA	5	88.8%	90.3%
DyDA	5	80.6%	83.6%

5. Discussion

Our results on three benchmark datasets—SwDA, MRDA, and DyDA—show that our model changes consistently improved dialogue act classification. The two main changes were: (1) replacing the BiGRU encoder with a Transformer encoder, and (2) improving speaker turn embeddings by adding labels for "speaker continued" and "speaker switched." These changes led to better performance across different types of conversations.

The biggest improvements were on MRDA and DyDA. These datasets include formal meetings and two-person daily conversations. In these cases, changes in speaker often mean a change in topic or intent. Our turn-aware embeddings helped the model use this information. Also, the Transformer's self-attention made it easier to learn long-range relationships, which is especially useful for MRDA where conversations are longer and more complex.

In the SwDA dataset, the improvement was smaller but still helpful. SwDA includes phone conversations with many types of labels (43 classes) and is less structured. Even in this noisier data, speaker change information helped. But this dataset might also need extra semantic signals to improve performance further.

Another important point is that our model trained smoothly and did not overfit. In all datasets, the validation and test accuracy were close, which shows the model learned in a stable and general way.

We also want to highlight that we kept all other parts of the system the same as the original baseline from He et al. (2021). This includes the dataset splits, input format, loss function, and training setup. Because of this, we can say that the improvements came from our two changes: using a Transformer and turn-aware speaker embeddings.

Finally, since we kept the RoBERTa-based sentence encoding and only added small changes, our model can still work well with future extensions like CRFs, topic modeling, or multitask learning. It provides a strong and flexible starting point for future research in dialogue act classification.

6. Conclusion

In this project, we improved a RoBERTa-based dialogue act classification model by making two key changes: (1) we replaced the BiGRU sequence encoder with a Transformer encoder, and (2) we refined the speaker embeddings to show whether the speaker continued or switched. These changes were based on the idea that speaker changes and long-range context are important in understanding conversations with multiple turns.

We tested our model on three well-known datasets—SwDA, MRDA, and DyDA—and saw consistent improvements in accuracy. The biggest gains were in MRDA and DyDA, where speaker turns follow clearer patterns. Even in the more complex and noisy SwDA dataset, our model performed better than the BiGRU-based version. This shows that using self-attention and speaker turn signals helps improve classification.

We kept all other parts of the original model the same—such as training setup, input format, and evaluation steps—to make sure the improvements came only from our changes. This makes our results fair and easy to reproduce.

Our work also creates opportunities for future research. For example, future models could use more global dialogue features, coherence information, or hierarchical structures. Our turn-aware embeddings might also help in other dialogue tasks like intent recognition or response generation.

In summary, we showed that modeling speaker transitions and using a Transformer encoder can make dialogue act classification more accurate. This highlights the importance of using structure-aware methods in conversational AI.

Limitations

While our improvements showed good results across all datasets, there are still some limitations that future work could address:

- **Limited Scope of Model Changes:** We kept most parts of the original model (like loss function, input format, and chunking) the same to clearly test only the effect of replacing BiGRU with Transformer and adding turn-aware embeddings. This helped us make fair comparisons, but we did not test other useful ideas like CRF decoding, multitask learning, or hierarchical models.
- **Simple Speaker Turn Representation:** Our model only uses a basic turn embedding to show whether the speaker continued or changed. It does not capture more detailed situations like interruptions or long pauses. A more advanced speaker model might help capture deeper interaction patterns.
- **No Pretraining for Turn Embeddings:** The speaker turn embeddings were randomly initialized and trained from scratch. This may limit their power, especially in small datasets. Pretraining or using helper tasks could make them more robust.
- **Only Accuracy Used for Evaluation:** We used accuracy as our main metric. This works well for balanced datasets, but it may not give a full picture for datasets like SwDA with many classes. Using more metrics (like per-class F1 score or confusion matrices) could provide deeper analysis.
- **Fixed Chunking Method:** We used the same chunking method as in the original model to divide long conversations. But this method does not always match the natural breaks in dialogue. Using a more flexible or dynamic chunking method might help keep context better.
- **Training Time Limits:** Because of time and hardware limits, we could not train for a long time. We report results using 100 epochs, but early stopping and testing were often done earlier. This could affect how reliable some final results are.

Even with these limits, our approach is simple and easy to build on. The improvements we saw across all datasets suggest that our method can be a strong base for future work in dialogue act classification.

References

[1] Chen, Q., Zhu, X., Ling, Z., Wei, S., Jiang, H., & Inkpen, D. (2017). Dialogue Act Recognition via CRF-Attentive Structured Network. *arXiv preprint.* arXiv:1711.05568.

[2] He, Zihao, Tavabi, Leili, Lerman, Kristina, and Soleymani, Mohammad.
Speaker Turn Modeling for Dialogue Act Classification.
Findings of the Association for Computational Linguistics: EMNLP 2021, pages 2150–2157, 2021.
https://aclanthology.org/2021.findings-emnlp.185

[3] Żelasko, P., Raj, D., & Hermansky, H. (2021). What Helps Transformers Recognize Conversational Structure? Importance of Context, Punctuation, and Labels in Dialog Act Recognition. *arXiv preprint*. arXiv:2107.02294.